Nama = Muhammad Wafiq Kamaluddin

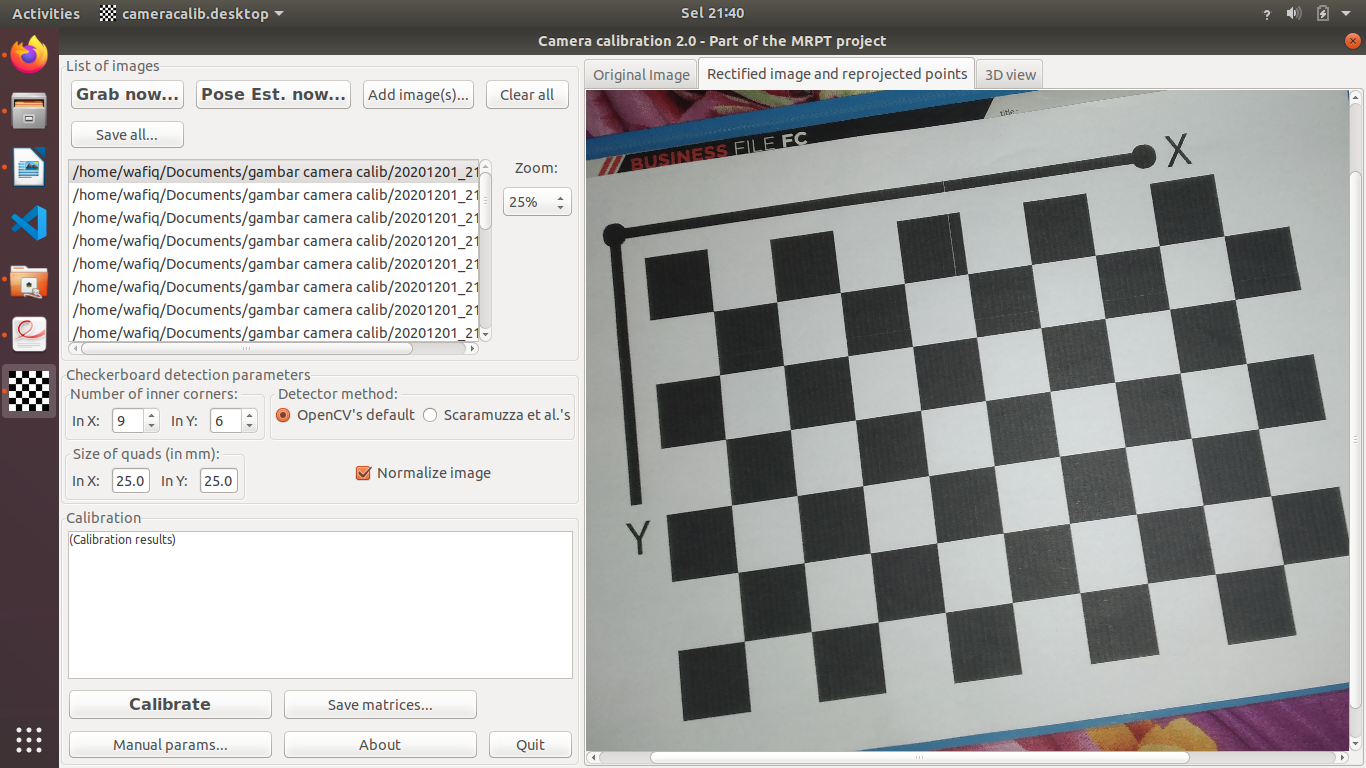
Nrp = 2210181042

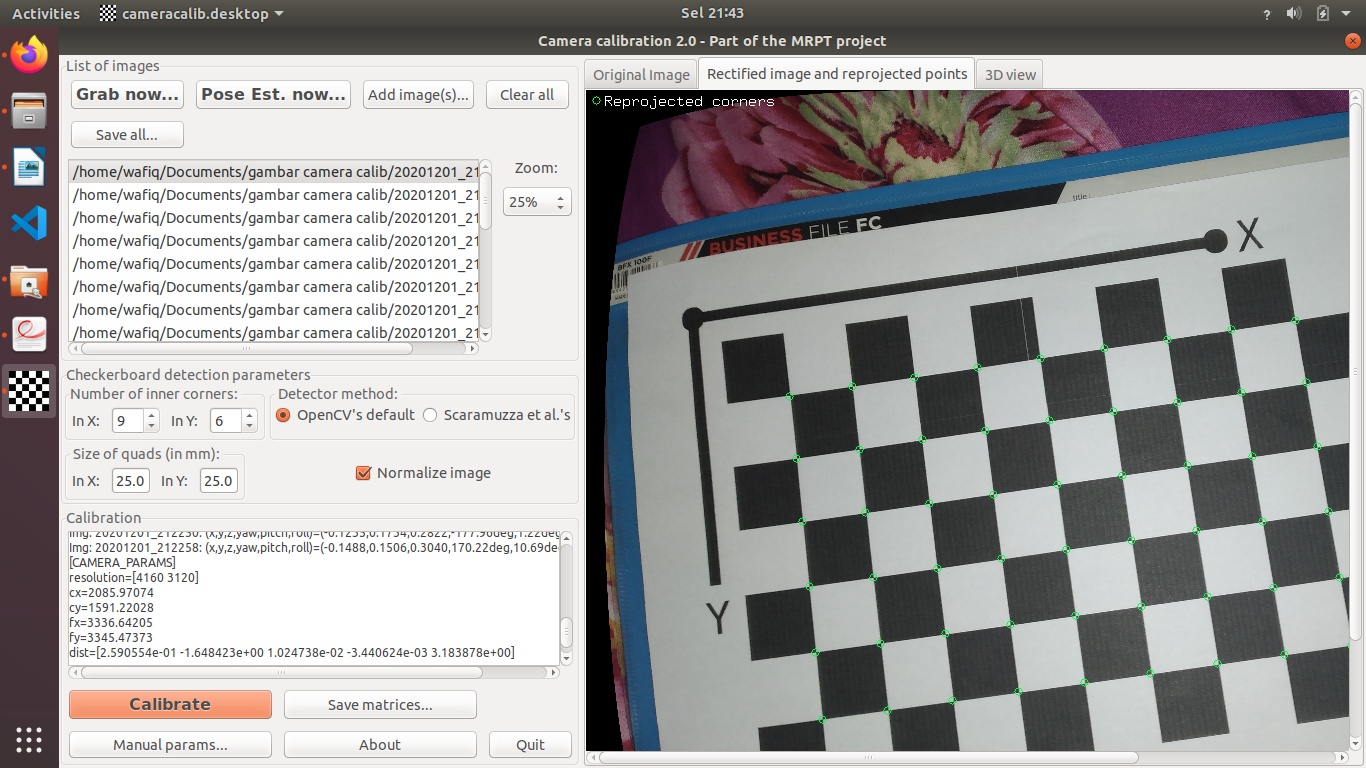
Kelas = 3 D4 TKB

pada percobaan kalibrasi kamera, saya menggunakan tools MPRT yaitu tools untuk penggembangan mobile robotik pada ubuntu.

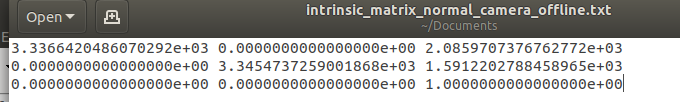
1. percobaan 1

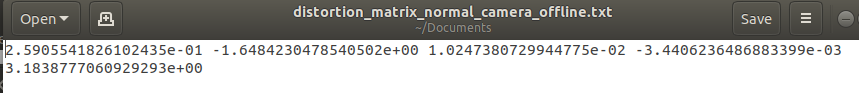
Offline

saya tekan kalibrate

pada text box akan muncul keterangan pada proses calibrate dan parameter kamera hasil dari kalibrasi

Matriks instrinsik kamera

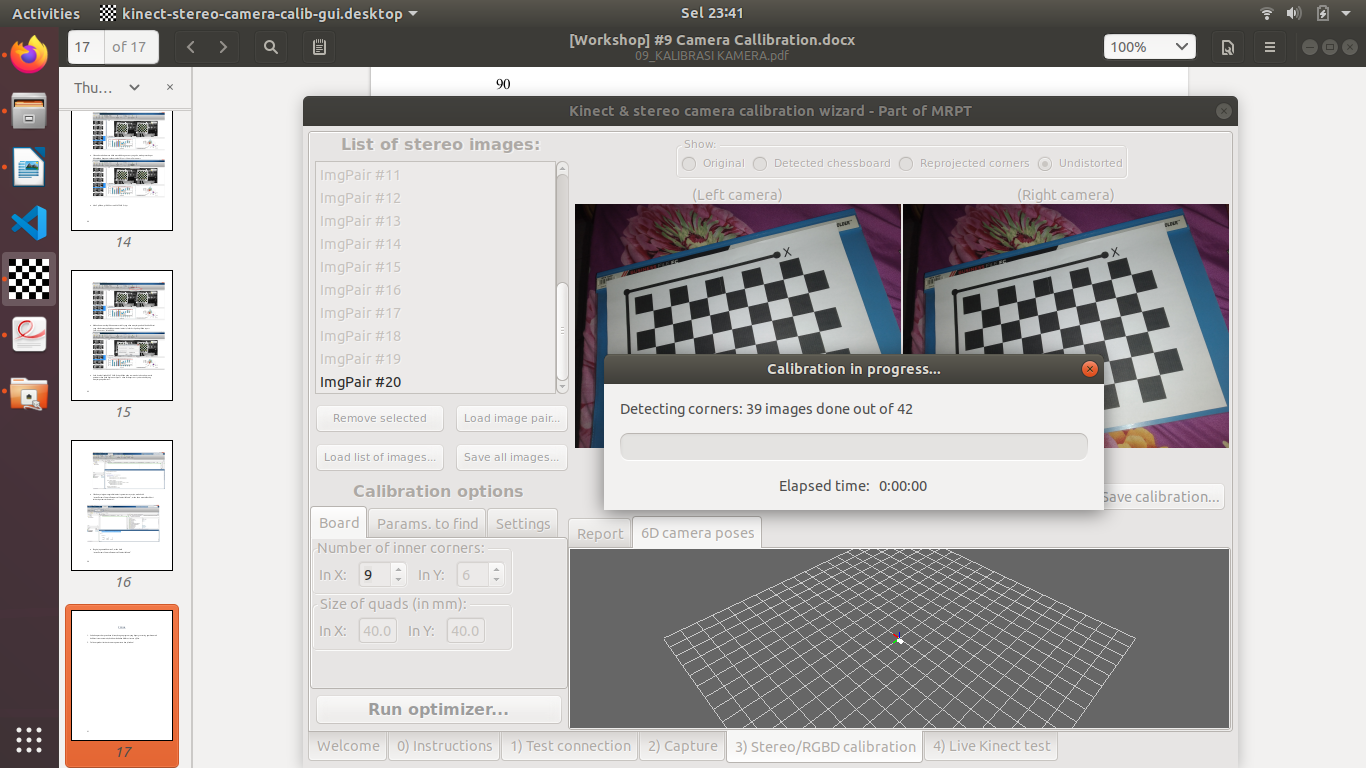
Matriks distorsi kamera

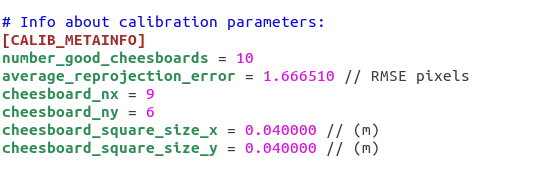
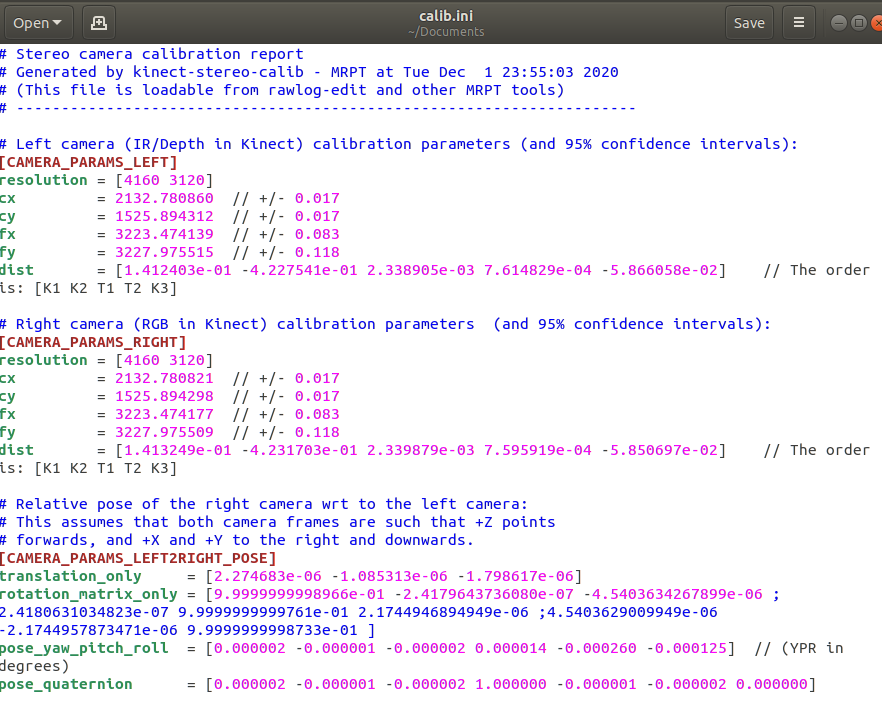


Online

2. Pecrobaan 2

Offline

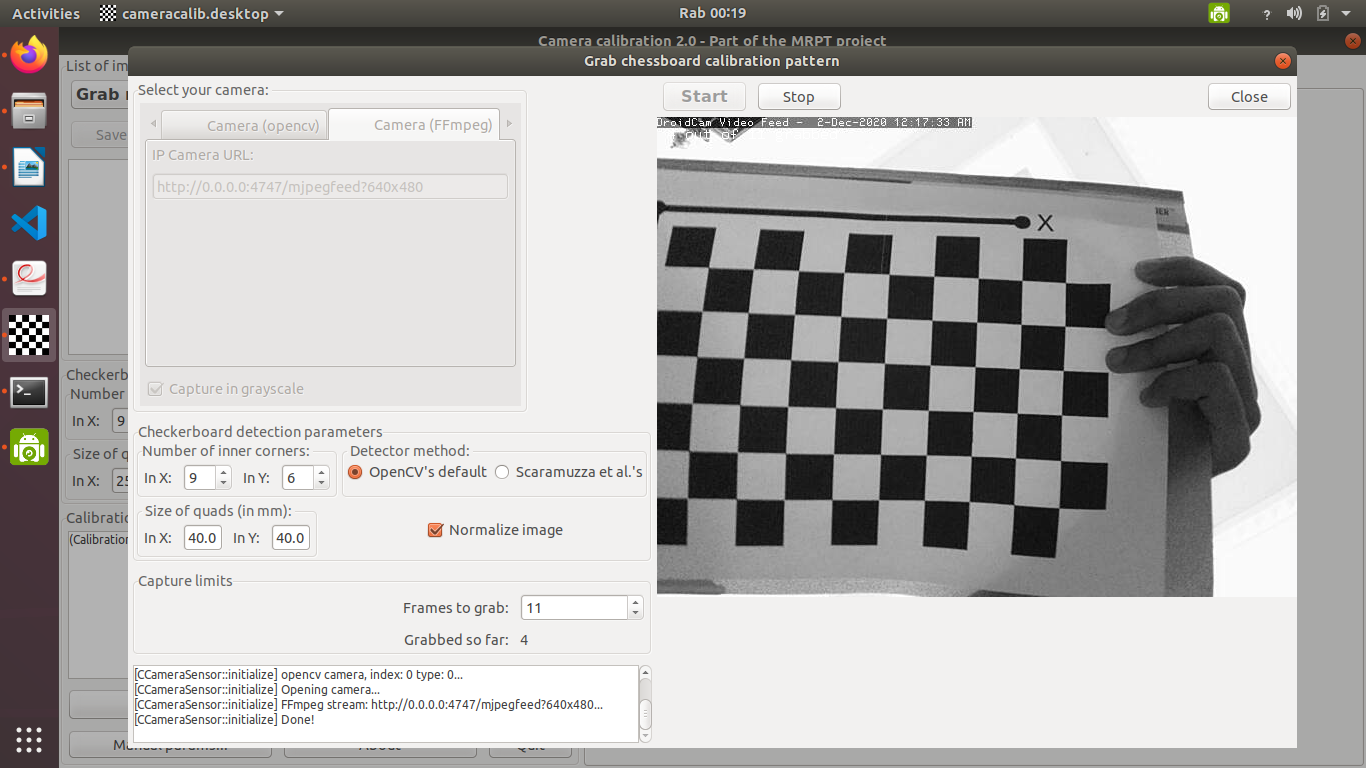
alurnya sama, saya inputkan terlebih dahulu gambar untuk kamera kanan dan kiri kemudian saya klik opimizer. Hasil dari kalibrasi adalah sebagai berikut

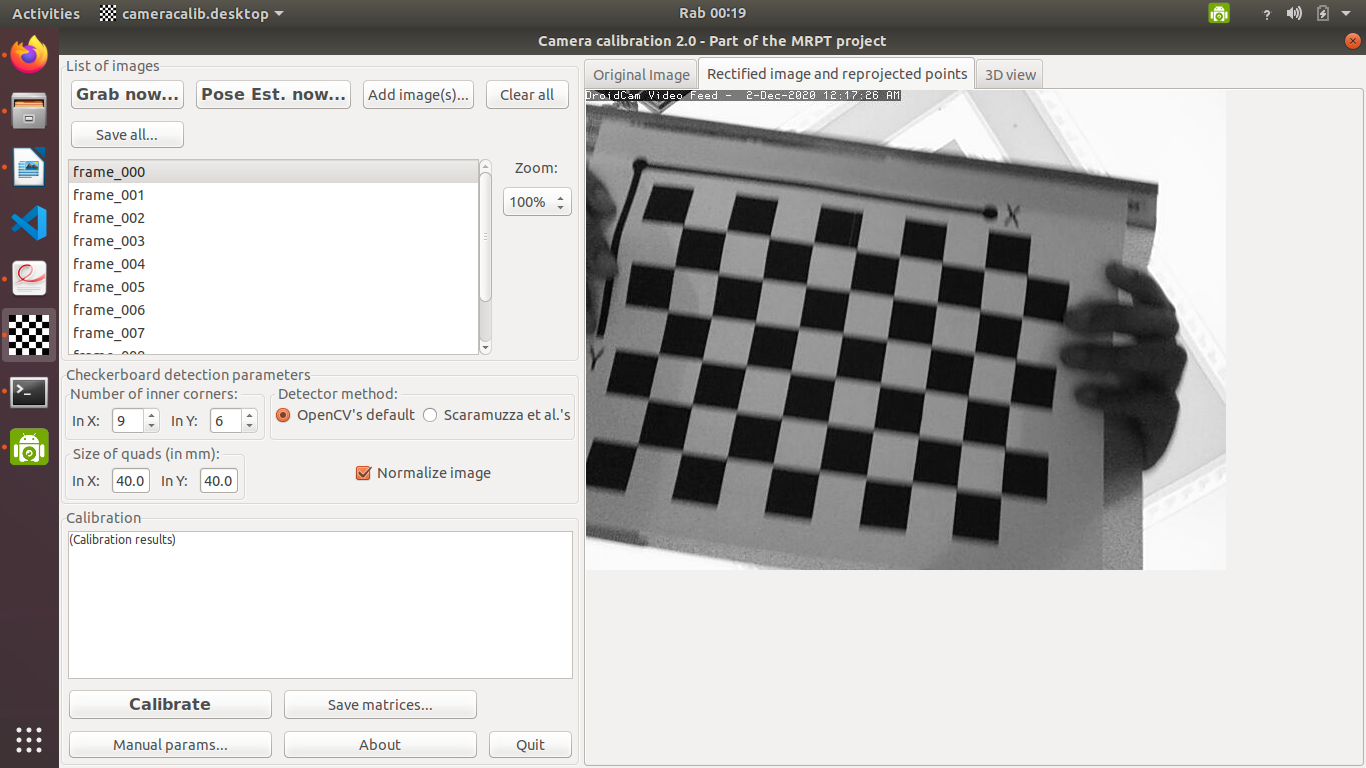


Online

kamera menggunakan kamera handphone melalui droidcam

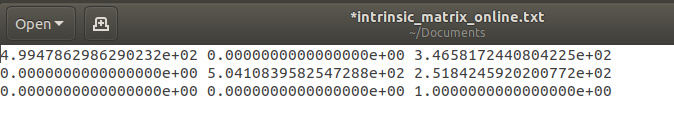
pengambilan gambar





hasil

matriks intrinsik



tugas

1.

kode

#include <opencv2/opencv.hpp>

#include <opencv2/calib3d/calib3d.hpp>

#include <opencv2/highgui/highgui.hpp>

#include <opencv2/imgproc/imgproc.hpp>

#include <stdio.h>

#include <iostream>

// Defining the dimensions of checkerboard

*int* CHECKERBOARD[2]{6,9};

*int* main()

{

// Creating vector to store vectors of 3D points for each checkerboard image

std::vector<std::vector<cv::Point3f> > objpoints;

// Creating vector to store vectors of 2D points for each checkerboard image

std::vector<std::vector<cv::Point2f> > imgpoints;

// Defining the world coordinates for 3D points

std::vector<cv::Point3f> objp;

for(*int* i{0}; i<CHECKERBOARD[1]; i++)

{

for(*int* j{0}; j<CHECKERBOARD[0]; j++)

objp.push\_back(cv::Point3f(j,i,0));

}

// Extracting path of individual image stored in a given directory

std::vector<cv::String> images;

// Path of the folder containing checkerboard images

std::string path = "/home/wafiq/Documents/gambar camera calib/\*.jpg";

cv::glob(path, images);

cv::Mat frame, gray;

// vector to store the pixel coordinates of detected checker board corners

std::vector<cv::Point2f> corner\_pts;

*bool* success;

// Looping over all the images in the directory

for(*int* i{0}; i<images.size(); i++)

{

frame = cv::imread(images[i]);

cv::cvtColor(frame,gray,cv::COLOR\_BGR2GRAY);

// Finding checker board corners

// If desired number of corners are found in the image then success = true

success = cv::findChessboardCorners(gray, cv::Size(CHECKERBOARD[0], CHECKERBOARD[1]), corner\_pts, CV\_CALIB\_CB\_ADAPTIVE\_THRESH | CV\_CALIB\_CB\_FAST\_CHECK | CV\_CALIB\_CB\_NORMALIZE\_IMAGE);

/\*

\* If desired number of corner are detected,

\* we refine the pixel coordinates and display

\* them on the images of checker board

\*/

if(success)

{

cv::TermCriteria criteria(CV\_TERMCRIT\_EPS | CV\_TERMCRIT\_ITER, 30, 0.001);

// refining pixel coordinates for given 2d points.

cv::cornerSubPix(gray,corner\_pts,cv::Size(11,11), cv::Size(-1,-1),criteria);

// Displaying the detected corner points on the checker board

cv::drawChessboardCorners(frame, cv::Size(CHECKERBOARD[0], CHECKERBOARD[1]), corner\_pts, success);

objpoints.push\_back(objp);

imgpoints.push\_back(corner\_pts);

}

cv::imshow("Image",frame);

cv::waitKey(0);

}

cv::destroyAllWindows();

cv::Mat cameraMatrix,distCoeffs,R,T;

/\*

7

\* Performing camera calibration by

7

\* passing the value of known 3D points (objpoints)

8

\* and corresponding pixel coordinates of the

8

\* detected corners (imgpoints)

8

\*/

cv::calibrateCamera(objpoints, imgpoints, cv::Size(gray.rows,gray.cols), cameraMatrix, distCoeffs, R, T);

std::cout << "cameraMatrix : " << cameraMatrix << std::endl;

std::cout << "distCoeffs : " << distCoeffs << std::endl;

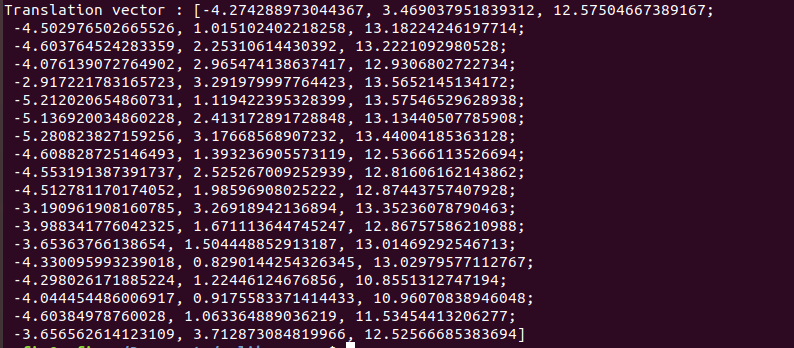
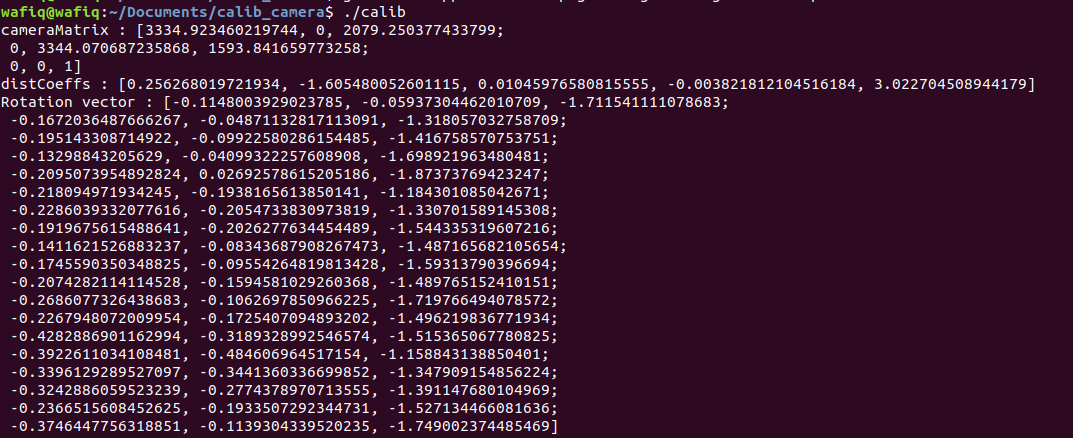
std::cout << "Rotation vector : " << R << std::endl;

std::cout << "Translation vector : " << T << std::endl;

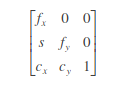
return 0;

}

Hasil



2. parameter instrinsik kamera adalah sebuah matriks yang dihasilkan dari pengolahan gambar yang berada pada koordinat kamera ke dalam gambar pada koordinat 2 D. parameter instrinsik kamera terdiri dari sebuah matriks ukuran 3x3 dengan bentuk umum sebagai berikut



fx dan fy adalah focal length dalam pixel. Focal length sendiri adalah parameter kamera beruba jarak lensa dengan jarak pengindra(“mata”) pada kamera.

S adalah koefisien skew, yaitu koefisien yang muncul akibat adanya distorsi gambar pada camera ruang kamera terhadap ruang dunia.

Cx dan cy adalah titik pada gambar dimana mata(“sensor”) terletak

untuk parameter intrinsik sudah ditampilkan pada percobaan diatas.